

\*\*\*

NNN	NNN		CCCCCCCCCCCC	NNN	NNN	FFFFFFF
NNN	NNN		CCCCCCCCCCCC	NNN	NNN	FFFFFFF
NNN	NNN		CCCCCCCCCCCC	NNN	NNN	FFF
NNN	NNN		CCC	NNN	NNN	FFF
NNN	NNN		CCC	NNN	NNN	FFF
NNN	NNN		CCC	NNN	NNN	FFF
NNNNNN	NNN		CCC	NNNNNN	NNN	FFF
NNNNNN	NNN		CCC	NNNNNN	NNN	FFF
NNNNNN	NNN		CCC	NNNNNN	NNN	FFF
NNN NNN NNN	NNN		CCC	NNN NNN NNN	NNN	FFFFF
NNN NNN NNN	NNN		CCC	NNN NNN NNN	NNN	FFFFF
NNN NNN NNN	NNN		CCC	NNN NNN NNN	NNN	FFFFF
NNN NNNNNN	NNN		CCC	NNN NNNNNN	NNN	FFF
NNN NNNNNN	NNN		CCC	NNN NNNNNN	NNN	FFF
NNN NNNNNN	NNN		CCC	NNN NNNNNN	NNN	FFF
NNN NNN	NNN		CCC	NNN NNN	NNN	FFF
NNN NNN	NNN		CCC	NNN NNN	NNN	FFF
NNN NNN	NNN		CCC	NNN NNN	NNN	FFF
NNN NNN	NNN		CCCCCCCC	NNN	NNN	FFF
NNN NNN	NNN		CCCCCCCC	NNN	NNN	FFF
NNN NNN	NNN		CCCCCCCC	NNN	NNN	FFF

CCCCCCCC	NN	NN	FFFFFFFFF		NN	NN	TTTTTTTTT	RRRRRRR	PPPPPPP	TTTTTTTTT
CCCCCCCC	NN	NN	FFFFFFFFF		NN	NN	TTTTTTTTT	RRRRRRR	PPPPPPP	TTTTTTTTT
CC	NN	NN	FF		NN	NN	TT	RR	RR	PP
CC	NN	NN	FF		NN	NN	TT	RR	RR	PP
CC	NNNN	NN	FF		NNNN	NN	TT	RR	RR	PP
CC	NNNN	NN	FF		NNNN	NN	TT	RR	RR	PP
CC	NN NN	NN	FFFFFFF		NN NN	NN	TT	RRRRRRR	PPPPPPP	TT
CC	NN NN	NN	FFFFFFF		NN NN	NN	TT	RRRRRRR	PPPPPPP	TT
CC	NN NNNN	FF			NN NNNN	NN	TT	RR RR	PP	TT
CC	NN NNNN	FF			NN NNNN	NN	TT	RR RR	PP	TT
CC	NN NNN	FF			NN NNN	NN	TT	RR RR	PP	TT
CC	NN NNN	FF			NN NNN	NN	TT	RR RR	PP	TT
CC	NN NNN	FF			NN NNN	NN	TT	RR RR	PP	TT
CC	NN NNN	FF			NN NNN	NN	TT	RR RR	PP	TT
CCCCCCCC	NN	NN	FF		NN	NN	TT	RR RR	PP	TT
CCCCCCCC	NN	NN	FF		NN	NN	TT	RR RR	PP	TT

LL		SSSSSSS
LL		SSSSSSS
LL		SS
LLLLLLLLL		SSSSSSS
LLLLLLLLL		SSSSSSS

```
1 0001 0 XTITLE 'DECnet Ethernet Configurator Module'
2 0002 0 MODULE CNFINTRPT (
3 0003 0   LANGUAGE (BLISS32),
4 0004 0   IDENT = 'V04-000'
5 0005 0   )
6 0006 0   ****
7 0007 0   ****
8 0008 0   *
9 0009 0   * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 0   * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 0   * ALL RIGHTS RESERVED.
12 0012 0   *
13 0013 0   * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 0   * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 0   * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 0   * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 0   * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 0   * TRANSFERRED.
19 0019 0   *
20 0020 0   * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 0   * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 0   * CORPORATION.
23 0023 0   *
24 0024 0   * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 0   * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 0   *
27 0027 0   *
28 0028 0   ****
29 0029 0   *
30 0030 0   *
31 0031 0   ++
32 0032 0   FACILITY: DECnet Configurator Module (NICONFIG)
33 0033 0   *
34 0034 0   ABSTRACT:
35 0035 0   *
36 0036 0   This module contains the routines for establishing, and breaking
37 0037 0   logical links to NICONFIG.
38 0038 0   *
39 0039 0   ENVIRONMENT: VAX/VMS Operating System
40 0040 0   *
41 0041 0   AUTHOR: Bob Grosso, CREATION DATE: 13-Oct-1982
42 0042 0   *
43 0043 0   MODIFIED BY:
44 0044 0   *
45 0045 0   V03-001 RPG0001 Bob Grosso 02-May-1983
46 0046 0   Ensure NICONFIG will die gracefully on Network Shutdown.
47 0047 0   --
48 0048 1 BEGIN
```

```
50      0049 1 XSBTTL 'Definitions'
51      0050 1
52      0051 1 ! INCLUDE FILES:
53      0052 1
54      0053 1
55      0054 1
56      0055 1 LIBRARY 'SYSSLIBRARY:STARLET'; ! VMS common definitions
57      0056 1
58      0057 1 LIBRARY 'SHRLIBS:NET'; ! Network definitions
59      0058 1
60      0059 1 LIBRARY 'SHRLIBS:NMALIBRY'; ! NICE Code definitions
61      0060 1
62      0061 1 REQUIRE 'LIBS:CNFDEF.R32';
63      0152 1
64      0153 1 REQUIRE 'SRC$:CNFPREFIX.REQ';
65      0250 1
66      0251 1
67      0252 1 ! BUILTIN functions
68      0253 1
69      0254 1 !
70      0255 1
71      0256 1 BUILTIN
72      0257 1     INSQUE,           ! INSQUE instruction
73      0258 1     REMQUE;          ! REMQUE instruction
74      0259 1
75      0260 1 !
76      0261 1     Own storage
77      0262 1 !
78      0263 1
79      0264 1 OWN
80      0265 1     SI_IOSB : BBLOCK [8]; ! IO status block for
81      0266 1
82      0267 1 !
83      0268 1 ! TABLE OF CONTENTS:
84      0269 1 !
85      0270 1
86      0271 1 FORWARD ROUTINE
87      0272 1
88      0273 1     CNF$SOLICIT_INTERRUPT : NOVALUE,        ! Solicit work items
89      0274 1     NET_INTERRUPT : NOVALUE,        ! Action routine to receive first SET command
90      0275 1     OPEN_REQUEST_LINK : NOVALUE,       ! Open incoming request logical link
91      0276 1     CNF$CLOSE_REQUEST_LINK : NOVALUE,    ! Close request logical link
92      0277 1     SHUTDOWN : NOVALUE,        ! Shut down receiver gracefully
93      0278 1     CNF$SOLICIT REQUEST : NOVALUE,       ! Perform a read on the logical link
94      0279 1     REQUEST_RECEIVED : NOVALUE;       ! Accept incoming request record
95      0280 1
96      0281 1
97      0282 1
98      0283 1 !
99      0284 1 ! EXTERNAL REFERENCES:
100     0285 1 !
101     0286 1
102     0287 1 EXTERNAL ROUTINE
103     0288 1
104     0289 1     ! Module CNFMAIN
105     0290 1
106     0291 1     CNF$EXIT,          ! Clean up and exit
```

107 0292 1 CNF\$TRACE, ! Log messages to log file  
108 0293 1 CNFSLOG\_DATA, ! Log messages to log file  
109 0294 1 CNFSGET\_ZVM, ! Get zeroed virtual memory  
110 0295 1 CNFSFREE\_VM, ! Free virtual memory  
111 0296 1 ! Module CNFREQUES  
112 0297 1 ! Module CNFWORKQ  
113 0298 1 ! NOVALUE, ! Discontinue surveillance on specified circuit  
114 0299 1 ! NOVALUE, ! Parse and perform requested function  
115 0300 1 ! Module CNFWORKQ  
116 0301 1 ! NOVALUE, ! Add work to work queue  
117 0302 1 ! NOVALUE, ! Add work to work queue  
118 0303 1 ! EXTERNAL LITERAL  
119 0304 1 ! NOVALUE, ! Mailbox error  
120 0305 1 ! NOVALUE, ! Error assigning or deassigning channel  
121 0306 1 ! NOVALUE, ! Error on logical link  
122 0307 1 ! NOVALUE, ! Maximum mailbox message size  
123 0308 1 ! NOVALUE, ! Synchronous event flag number  
124 0309 1 ! NOVALUE, ! Asynchronous event flag number  
125 0310 1 ! EXTERNAL  
126 0311 1 ! NOVALUE, ! Mailbox message buffer  
127 0312 1 ! NOVALUE, ! Channel opened to network  
128 0313 1 ! NOVALUE, ! Channel to mailbox  
129 0314 1 ! NOVALUE, ! Boolean: mark if useful work is being accomplished  
130 0315 1 ! NOVALUE, ! Logging control mask  
131 0316 1 ! NOVALUE, ! List of circuit under surveillance  
132 0317 1 ! NOVALUE, ! Listhead for incoming links  
133 0318 1 ! NOVALUE, !  
134 0319 1 ! NOVALUE, !  
135 0320 1 ! NOVALUE, !  
136 0321 1 ! NOVALUE, !  
137 0322 1 ! NOVALUE, !  
138 0323 1 ! NOVALUE, !  
139 0324 1 ! NOVALUE, !  
140 0325 1 ! NOVALUE, !  
141 0326 1 ! NOVALUE, !  
142 0327 1 ! NOVALUE, !  
143 0328 1 ! NOVALUE, !

```

145 0329 1 %SBTTL 'CNF$SOLICIT_INTERRUPT Request network interrupts for Connect requests and Shutdown'
146 0330 1 GLOBAL ROUTINE CNF$SOLICIT_INTERRUPT: NOVALUE =
147 0331 1
148 0332 1 ++
149 0333 1 FUNCTIONAL DESCRIPTION:
150 0334 1
151 0335 1 Issue an asynchronous QIO on the associated mailbox
152 0336 1 for the network channel in expectation of receiving
153 0337 1 requests for connects, or Shutdown notification.
154 0338 1 Called the first time from MAIN routine in user mode
155 0339 1 and subsequent times from NET_INTERRUPT to execute in AST mode.
156 0340 1
157 0341 1 FORMAL PARAMETERS:
158 0342 1 NONE
159 0343 1
160 0344 1 IMPLICIT INPUTS:
161 0345 1
162 0346 1     CNFSW_MBXCHAN    Channel number for mailbox
163 0347 1     CNFSA_MBXMSG     Buffer for mailbox msg
164 0348 1
165 0349 1 IMPLICIT OUTPUTS:
166 0350 1 NONE
167 0351 1
168 0352 1 ROUTINE VALUE:
169 0353 1 COMPLETION CODES:
170 0354 1
171 0355 1 Errors are signalled
172 0356 1
173 0357 1 SIDE EFFECTS:
174 0358 1 NONE
175 0359 1
176 0360 1 --
177 0361 1
178 0362 2 BEGIN
179 0363 2
180 0364 2 LOCAL
181 0365 2 STATUS:
182 0366 2
183 0367 2 CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),
184 0368 2 $DESCRIPTOR ?'cnf$solicit_interrupt'));
185 0369 2
186 P 0370 2 STATUS = $QIO (
187 P 0371 2     FUNC = IOS_READVBLK, ! Request read on mailbox
188 P 0372 2     CHAN = .CNFSW_MBXCHAN, ! Use assigned channel
189 P 0373 2     EFN = CNFSC_ASYNCH_EFN, ! Asynchronous Event flag number
190 P 0374 2     IOSB = SI_IOSB, ! Interrupt request block
191 P 0375 2     ASTADR = RET_INTERRUPT, ! AST routine to execute on read completion
192 P 0376 2     P1 = CNFSA_MBXMSG, ! Buffer to contain mailbox message
193 P 0377 2     P2 = CNFSC_MAXMBXMSG); ! Size maximum on mailbox message
194 P 0378 2
195 P 0379 2 IF NOT .STATUS
196 P 0380 2 THEN ! report an error
197 P 0381 2     SIGNAL (CNFS_MAILBOX, 0, .STATUS);
198 P 0382 2
199 P 0383 2 RETURN:
200 P 0384 2 END; ! End routine CNF$SOLICIT_INTERRUPT

```

```

        .TITLE CNFINTRPT DECnet Ethernet Configurator Module
        .IDENT \V04-000\

        .PSECT SPLITS,NOWRT,NOEXE,2

        45 43 41 52 54 00000 P.AAB: .ASCII \TRACE\
        00005 .BLKB 3
        00000005 00008 P.AAA: .LONG 5
        00000000 0000C .ADDRESS P.AAB
        74 6E 69 5F 74 69 63 69 6C 6F 73 24 66 6E 63 00010 P.AAD: .ASCII \cnf$solicit_interrupt\
        74 70 75 72 72 65 0001F .BLKB 3
        00000015 00028 P.AAC: .LONG 21
        00000000 0002C .ADDRESS P.AAD

        .PSECT SOWNS,NOEXE,2

        00000 SI_IOSB:.BLKB 8

        .EXTRN CNFSEXIT, CNF$TRACE
        .EXTRN CNFSLOG DATA, CNF$GET ZVM
        .EXTRN CNFSFREE VM, CNF$DISABLE_SURVEIL
        .EXTRN CNF$PROCESS REQUEST
        .EXTRN WKQSADD WORK ITEM
        .EXTRN WKQSDO WORK ITEM
        .EXTRN CNFS_MAILBOX, CNFS_CHAN
        .EXTRN CNFS_LINK, CNFSC_MAXMBXMSG
        .EXTRN CNFSC_SYNCH_EFN
        .EXTRN CNFSC_ASYNC_EFN
        .EXTRN CNFSA_MBXMSG, CNFSW_NETCHAN
        .EXTRN CNFSW_MBXCHAN, CNFSB_SURVEILLANCE_SET
        .EXTRN CNFSGE_LOGMASK, CNFSGQ_CIRSLURLST
        .EXTRN CNFSGQ_IRBLST, SYSSQIO

        .PSECT SCODES,NOWRT,2

        0000G CF 0000 00000 .ENTRY CNFS$SOLICIT_INTERRUPT, Save nothing : 0330
        0000' CF 9F 00002 PUSHAB P.AAC : 0368
        0000' CF 9F 00006 PUSHAB P.AAA : 0367
        0000G CF 01 DD 0000A PUSHL #1
        0000G CF 03 FB 0000C CALLS #3, CNF$TRACE
        0000G CF 7E 7C 00011 CLRQ -(SP)
        0000G CF 7E 7C 00013 CLRQ -(SP) : 0377
        00000000G 8F DD 00015 PUSHL #CNFSC_MAXMBXMSG
        0000G CF 9F 0001B PUSHAB CNFSA_MBXMSG
        0000V CF 7E D4 0001F CLRL -(SP)
        0000V CF 9F 00021 PUSHAB NET_INTERRUPT
        0000' CF 9F 00025 PUSHAB SI_IOSB
        0000' CF 31 DD 00029 PUSHL #49
        0000G CF 7E 0000G MOVZWL CNFSW_MBXCHAN, -(SP)
        00000000G 8F DD 00030 PUSHL #CNFSC_ASYNC_EFN
        00000000G 00 0C FB 00036 CALLS #12, SYSSQIO
        11 50 E8 0003D BLBS STATUS, 1S : 0379
        50 DD 00040 PUSHL STATUS
        00000000G 7E D4 00042 CLRL -(SP)
        00000000G 8F DD 00044 PUSHL #CNFS_MAILBOX : 0381
    
```

CNFINTRPT  
V04-000

DECnet Ethernet Configurator Module  
CNFS\$SOLICIT\_INTERRUPT Request network interru

L 9  
16-Sep-1984 02:03:38  
14-Sep-1984 12:49:50

VAX-11 Bliss-32 V4.0-742  
[NICNF.SRC]CNFINTRPT.B32;1

Page 6  
(3)

00000000G 00      03 FB 0004A      CALLS #3, LIB\$SIGNAL  
                  04 00051 1\$:      RET

: 0384

: Routine Size: 82 bytes.   Routine Base: \$CODES + 0000

CNF  
V04

```

202      0385 1 XSBTTL 'net_interrupt Process Net interrupts for Shutdown or Connect Request'
203      0386 1 ROUTINE NET_INTERRUPT : NOVALUE =
204
205      0387 1 ++
206      0388 1 ++
207      0389 1 This AST routine is called when the outstanding QIO
208          on the associated mailbox completes. If the interrupt
209          indicates a connect is pending, then the acceptance
210          routine is added to the work queue.
211
212      0395 1 --
213
214      0396 1 BEGIN
215      0397 2 BIND
216          MESSAGE_TYPE = CNFSA_MBXMSG [0] : BYTE;           ! First byte contains code for message type
217
218      0401 2 CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),
219          $DESCRIPTOR ('Net_interrupt'));
220
221      0404 2
222          Check message type. If connect request, then connect and wait
223          for a set. Anything else is bad news.
224
225      0408 2 SELECTONEU .MESSAGE_TYPE OF
226          SET
227
228      0411 2 [MSG$_NETSHUT]:                                ! Network shutting down
229
230      0413 3 BEGIN
231          WKQ$ADD WORK_ITEM(SHUTDOWN);                   ! Shut down receiver gracefully
232          CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),
233              $DESCRIPTOR ('Net_interrupt - Network shutting down'));
234          RETURN;                                         ! Do not re-issue mailbox read
235          END;
236
237
238      0421 2 [MSG$ CONNECT]:                            ! Incoming connect request
239
240      0423 3 BEGIN
241          LOCAL
242              PTR,LEN,
243                  IRB:     REF BBLOCK;                 ! Incoming Request Block
244
245          CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),
246              $DESCRIPTOR ('Net_interrupt - Connect request received'));
247
248          LEN = IRB$C_LENGTH;
249          CNF$GET ZVM(LEN,IRB);
250          IRB [IRBSW_SIZE] = IRB$C_LENGTH;             ! Allocate incoming request block
251          PTR = 5 + CNFSA_MBXMSG [4];                 ! Set length of block
252          IRB [IRBSB_NCBLEN] = .CNFSA_MBXMSG [.PTR];   ! Get index of start of ascic data
253          CH$COPY (.CNFSA_MBXMSG [.PTR], CNFSA_MBXMSG [.PTR+1], ! Set length of NCB
254              0, IRB$C_MAXNCBLEN, IRB [IRB$T_NCB]);
255          IRB [IRBSL_BNR_F[INK]] = IRB [IRBSL_BNR_FLINK]; ! Initialize list for Buffered NICE Messages
256          IRB [IRBSL_BNR_BLINK] = IRB [IRBSL_BNR_FLINK];
257          INSQUE (.IRB, .CNF$GQ_IRBLST [1]);           ! Insert into list
258
259          WKQ$ADD_WORK_ITEM(OPEN_REQUEST_LINK,.IRB);    ! Queue the connect accept

```

```

259      0442 2      END;
260      0443 2
261      0444 2      [OTHERWISE]:
262      0445 2      BEGIN
263      0446 3      CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE *** ERROR'),
264      0447 3          $DESCRIPTOR T'Net_interrupt - Unprocessed Interrupt'));
265      0448 2      END;
266      0449 2
267      0450 2      TES;
268      0451 2
269      0452 2      CNF$SOLICIT_INTERRUPT();           ! Issue another read on mailbox
270      0453 2
271      0454 2      CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),
272      0455 2          $DESCRIPTOR T'Net_interrupt - Interrupt dispatched, another solicited'));
273      0456 2
274      0457 2      RETURN;
275      0458 1      END;                           ! Routine net_interrupt

```

			.PSECT SPLIT\$,NOWRT,NOEXE,2
		45 43 41 52 54 00030 P.AAF:	.ASCII \TRACE\
		00000005 00035 P.AAE:	.BLKB 3
		00000000 00038 P.AAE:	.LONG 5
		00000000 0003C P.AAF:	.ADDRESS P.AAF
		00040 P.AAH:	.ASCII \Net_interrupt\
		0004D P.AAH:	.BLKB 3
		00050 P.AAG:	.LONG 13
		00054 P.AAG:	.ADDRESS P.AAH
		00058 P.AAJ:	.ASCII \TRACE\
		0005D P.AAJ:	.BLKB 3
		00060 P.AAI:	.LONG 5
		00064 P.AAI:	.ADDRESS P.AAJ
		00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\
74 70 75 72 72 65 74 6E 49 5F 74 65 4E	45 43 41 52 54 00030 P.AAF:	.ASCII \TRACE\	
	00000005 00035 P.AAE:	.BLKB 3	
	00000000 00038 P.AAE:	.LONG 5	
	00000000 0003C P.AAF:	.ADDRESS P.AAF	
	00040 P.AAH:	.ASCII \Net_interrupt\	
	0004D P.AAH:	.BLKB 3	
	00050 P.AAG:	.LONG 13	
	00054 P.AAG:	.ADDRESS P.AAH	
	00058 P.AAJ:	.ASCII \TRACE\	
	0005D P.AAJ:	.BLKB 3	
	00060 P.AAI:	.LONG 5	
	00064 P.AAI:	.ADDRESS P.AAJ	
	00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\	
2D 20 74 70 75 72 72 65 74 6E 49 5F 74 65 4E	45 43 41 52 54 00030 P.AAF:	.ASCII \TRACE\	
69 74 74 75 68 73 20 6B 72 6F 77 74 65 4E 20	00000005 00035 P.AAE:	.BLKB 3	
	00000000 00038 P.AAE:	.LONG 5	
	00000000 0003C P.AAF:	.ADDRESS P.AAF	
	00040 P.AAH:	.ASCII \Net_interrupt\	
	0004D P.AAH:	.BLKB 3	
	00050 P.AAG:	.LONG 13	
	00054 P.AAG:	.ADDRESS P.AAH	
	00058 P.AAJ:	.ASCII \TRACE\	
	0005D P.AAJ:	.BLKB 3	
	00060 P.AAI:	.LONG 5	
	00064 P.AAI:	.ADDRESS P.AAJ	
	00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\	
	00080 P.AAL:	.BLKB 3	
	00085 P.AAL:	.LONG 37	
	00090 P.AAK:	.ADDRESS P.AAL	
	00094 P.AAK:	.ASCII \TRACE\	
	00098 P.AAN:	.BLKB 3	
	0009D P.AAN:	.LONG 5	
	000A0 P.AAM:	.ADDRESS P.AAN	
	000A4 P.AAP:	.ASCII \Net_interrupt - Connect request received\	
	000A8 P.AAP:	.BLKB 3	
2D 20 74 70 75 72 72 65 74 6E 49 5F 74 65 4E	45 43 41 52 54 00030 P.AAF:	.ASCII \TRACE\	
73 65 75 71 65 72 20 74 63 65 6E 6F 43 20	00000005 00035 P.AAE:	.BLKB 3	
	00000000 00038 P.AAE:	.LONG 5	
	00000000 0003C P.AAF:	.ADDRESS P.AAF	
	00040 P.AAH:	.ASCII \Net_interrupt\	
	0004D P.AAH:	.BLKB 3	
	00050 P.AAG:	.LONG 13	
	00054 P.AAG:	.ADDRESS P.AAH	
	00058 P.AAJ:	.ASCII \TRACE\	
	0005D P.AAJ:	.BLKB 3	
	00060 P.AAI:	.LONG 5	
	00064 P.AAI:	.ADDRESS P.AAJ	
	00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\	
	00080 P.AAL:	.BLKB 3	
	00085 P.AAL:	.LONG 40	
	00090 P.AAK:	.ADDRESS P.AAL	
	00094 P.AAK:	.ASCII \TRACE *** ERROR\	
	00098 P.AAN:	.BLKB 1	
	0009D P.AAN:	.LONG 15	
	000A0 P.AAM:	.ADDRESS P.AAN	
	000A4 P.AAP:	.ASCII \Net_interrupt - Connect request received\	
	000A8 P.AAP:	.BLKB 3	
52 4F 52 52 45 20 2A 2A 2A 20 45 43 41 52 54	00000005 00030 P.AAF:	.ASCII \TRACE\	
	00000000 00035 P.AAE:	.BLKB 3	
	00000000 00038 P.AAE:	.LONG 5	
	00000000 0003C P.AAF:	.ADDRESS P.AAF	
	00040 P.AAH:	.ASCII \Net_interrupt\	
	0004D P.AAH:	.BLKB 3	
	00050 P.AAG:	.LONG 13	
	00054 P.AAG:	.ADDRESS P.AAH	
	00058 P.AAJ:	.ASCII \TRACE\	
	0005D P.AAJ:	.BLKB 3	
	00060 P.AAI:	.LONG 5	
	00064 P.AAI:	.ADDRESS P.AAJ	
	00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\	
	00080 P.AAL:	.BLKB 3	
	00085 P.AAL:	.LONG 40	
	00090 P.AAK:	.ADDRESS P.AAL	
	00094 P.AAK:	.ASCII \TRACE *** ERROR\	
	00098 P.AAN:	.BLKB 1	
	0009D P.AAN:	.LONG 15	
	000A0 P.AAM:	.ADDRESS P.AAN	
	000A4 P.AAP:	.ASCII \Net_interrupt - Connect request received\	
	000A8 P.AAP:	.BLKB 3	
2D 20 74 70 75 72 72 65 74 6E 49 5F 74 65 4E	45 43 41 52 54 00030 P.AAF:	.ASCII \TRACE\	
6E 49 20 64 65 73 73 65 74 6F 72 75 72 72 65	00000005 00035 P.AAE:	.BLKB 3	
	00000000 00038 P.AAE:	.LONG 5	
	00000000 0003C P.AAF:	.ADDRESS P.AAF	
	00040 P.AAH:	.ASCII \Net_interrupt\	
	0004D P.AAH:	.BLKB 3	
	00050 P.AAG:	.LONG 13	
	00054 P.AAG:	.ADDRESS P.AAH	
	00058 P.AAJ:	.ASCII \TRACE\	
	0005D P.AAJ:	.BLKB 3	
	00060 P.AAI:	.LONG 5	
	00064 P.AAI:	.ADDRESS P.AAJ	
	00068 P.AAL:	.ASCII \Net_interrupt - Network shutting down\	
	00080 P.AAL:	.BLKB 3	
	00085 P.AAL:	.LONG 40	
	00090 P.AAK:	.ADDRESS P.AAL	
	00094 P.AAK:	.ASCII \TRACE *** ERROR\	
	00098 P.AAN:	.BLKB 1	
	0009D P.AAN:	.LONG 15	
	000A0 P.AAM:	.ADDRESS P.AAN	
	000A4 P.AAP:	.ASCII \Net_interrupt - Connect request received\	
	000A8 P.AAP:	.BLKB 3	
	00090 P.AAK:	.ASCII \Net_interrupt - Unprocessed Interrupt\	
	00094 P.AAK:	.BLKB 3	

.PSECT SCODES,NOWRT,2

03FC 00000 NET_INTERRUPT:									
59	0000G	CF	9E	00002	.WORD	Save R2,R3,R4,R5,R6,R7,R8,R9			0386
58	0000G	CF	9E	00007	MOVAB	CNF\$TRACE, R9			
57	0000'	CF	9E	0000C	MOVAB	MESSAGE_TYPE, R8			
5E		08	C2	00011	MOVAB	P.AAG, R7			
		57	DD	00014	SUBL2	#8, SP			
		E8	A7	9F	PUSHL	R7			
			01	DD	PUSHAB	P.AAE			0402
69			03	FB	PUSHL	#1			0401
50			68	9A	CALLS	#3, CNF\$TRACE			0408
38			50	91	MOVZBL	MESSAGE_TYPE, R0			0411
			11	12	CMPB	R0, #59			
			12	00024	BNEQ	1\$			
0000G	CF	0000V	CF	9F	PUSHAB	SHUTDOWN			0414
			01	FB	CALLS	#1, WK\$ADD_WORK_ITEM			
			40	A7	PUSHAB	P.AAK			0416
			10	A7	PUSHAB	P.AAI			0415
			7B	11	BRB	4\$			
		32	50	91	CMPB	R0, #50			0421
			50	91	BNEQ	2\$			
			5C	12	PUSHAB	P.AAO			0428
			0080	C7	PUSHAB	P.AAM			0427
			50	A7	PUSHAB	#1			
			01	DD	PUSHL	#3, CNF\$TRACE			0430
04	69	015F	03	FB	CALLS	#351, LEN			0431
	AE		8F	3C	MOVZWL	SP			
			5E	DD	PUSHL	LEN			
			08	AE	PUSHAB	#2, CNF\$GET_ZVM			
0000G	CF	015F	02	FB	CALLS	IRB, R6			0432
	56		6E	D0	MOVZBL	#351, 8(R6)			
08	A6	015F	8F	B0	MOVW	CNFSA_MBXMSG+4, PTR			0433
	50		04	A8	MOVZBL	#5, PTR			
	50		05	C0	ADDL2	CNFSA_MBXMSG[PTR], 36(R6)			0434
24	A6		6840	90	MOVBL	CNFSA_MBXMSG[PTR], R1			0435
	51		6840	9A	MOVZBL	R1, CNFSA_MBXMSG+[PTR], #0, #64, 37(R6)			0436
0040	8F	00	01	A840	MOVCS				
			25	A6	0007A				
			14	A6	MOVAB	20(R6), 20(R6)			0437
			18	A6	MOVAB	20(R6), 24(R6)			0438
			0000G	DF	INSQUE	(R6), ACNF\$GQ_IRBLST+4			0439

CNFINTRPT  
V04-000

DECnet Ethernet Configurator Module  
net\_interrupt Process Net interrupts for Shutt

{ 10  
16-Sep-1984 02:03:38  
14-Sep-1984 12:49:50

VAX-11 Bliss-32 V4.0-742  
[NICNF.SRC]CNFINTRPT.B32;1

Page 10  
(4)

CN  
VO

		6E	DD	0008B	PUSHL	IRB		: 0441
		0000V	CF	9F 0008D	PUSHAB	OPEN REQUEST_LINK		
	0000G CF		02	FB 00091	CALLS	#2, QKQSADD_WORK_ITEM		
			0D	11 00096	BRB	38		: 0408
		00C8	C7	9F 00098	2\$:	PUSHAB	P.AAS	: 0447
		0098	C7	9F 0009C	PUSHAB	P.AAQ		: 0446
			01	DD 000A0	PUSHL	#1		
	FF04 69		03	FB 000A2	CALLS	#3, CNF\$TRACE		
			00	FB 000A5	3\$:	CALLS	#0, CNF\$SOLICIT_INTERRUPT	: 0452
		0118	C7	9F 000AA	PUSHAB	P.AAW		: 0455
		00D8	C7	9F 000AE	PUSHAB	P.AAU		: 0454
			01	DD 000B2	4\$:	PUSHL	#1	
	69		03	FB 000B4	CALLS	#3, CNF\$TRACE		
			04	000B7	RET			: 0458

; Routine Size: 184 bytes, Routine Base: \$CODE\$ + 0052



```

334      0516 2 THEN
335      0517 BEGIN
336      0518 SIGNAL (CNFS_CHAN, 0, .STATUS);
337      0519 CNF$CLOSE_REQUEST_LINK(.IRB);
338      0520 RETURN;
339      0521 END;
340
341      P 0522 STATUS = $QIOW(FUNC = IOS_ACCESS,
342          CHAN = .IRB [IRBSW_CHAN],
343          EFN = CNF$C_SYNCH_EFN,
344          IOSB = IRB [IRBSW_IOSB],
345          P2 = NCB_DESC);
346
347      P 0523 IF .STATUS
348      P 0524 THEN
349      P 0525 STATUS = .IRB [IRBSW_IOSB];
350
351      P 0526 IF NOT .STATUS
352      P 0527 THEN
353      P 0528 BEGIN
354          SIGNAL (CNFS_LINK, 0, .STATUS);
355          CNF$CLOSE_REQUEST_LINK(.IRB);
356          RETURN;
357          END;
358
359      P 0529 CNF$SOLICIT_REQUEST (.IRB);
360
361      P 0530 1 END;
362
                                         ! Accept the logical link
                                         ! Address of I/O status block
                                         ! Address of network control block
                                         ! If successfully submitted,
                                         ! then pick up QIO final status
                                         ! If error starting up link
                                         ! then report the error
                                         ! and deallocate the storage
                                         ! Issue a QIO for an incoming request record
                                         ! End routine open_request_link

```

```

.PSECT SPLIT$,NOWRT,NOEXE,2
45 43 41 52 54 00170 P.AAZ: .ASCII \TRACE\
                               00175 .BLKB 3
                               00000005. 00178 P.AAY: .LONG 5
                               00000000. 0017C .ADDRESS P.AAZ
                               00180 P.ABB: .ASCII \open_request_link\
                               0018F
                               00191 .BLKB 3
                               00000011. 00194 P.ABA: .LONG 17
                               00000000. 00198 .ADDRESS P.ABB
                               0019C P.ABD: .ASCII \NET:\<0><0><0>
                               010E0005. 001A4 P.ABC: .LONG 17694725
                               00000000. 001A8 .ADDRESS P.ABD
                               .EXTRN SY$ASSIGN, SY$QIOW
.PSECT SCODE$,NOWRT,2
000C 00000 OPEN_REQUEST_LINK:
5E      0000. 08 C2 00002 .WORD Save R2,R3
                               CF 9F 00005 .SUBL2 #8, SP
                               0000. CF 9F 00009 .PUSHAB P.ABA
                               01 DD 0000D .PUSHAB P.AAY
                               .PUSHL #1

```

0460  
0487  
0486

CNF INTRPT  
V04-000

## DECnet Ethernet Configurator Module open\_request\_link

F 10  
16-Sep-1984 02:03:38 VAX-11 Bliss-32 v4.0-742  
14-Sep-1984 12:49:50 [NICNF.SRC]CNF INTRPT.B32;1

Page 13  
(5)

; Routine Size: 175 bytes,      Routine Base: \$CODE\$ + 010A

```
364      0545 1 %SBTTL 'CNFS CLOSE REQUEST LINK'  
365      0546 1 GLOBAL ROUTINE CNFS CLOSE REQUEST LINK (IRB): NOVALUE =  
366  
367  
368  
369      0547 1 ---  
370      0548 1 Close the logical link for incoming request records.  
371      0549 1 Inputs:  
372      0550 1     irb = Address of incoming request block  
373      0551 1 Outputs:  
374      0552 1     None  
375      0553 1 Value:  
376      0554 1     Signal any errors  
377      0555 1 ---  
378      0556 1 BEGIN  
379      0557 1 MAP  
380      0558 1     IRB:      REF BBLOCK;           ! Address of incoming request channel  
381      0559 1 LOCAL  
382      0560 1     FREE_BNR : REF BBLOCK,  
383      0561 1     LENGTH,  
384      0562 1     STATUS;  
385  
386      0563 1 BEGIN  
387      0564 1 MAP  
388      0565 1     IRB:      REF BBLOCK;           ! Address of incoming request channel  
389      0566 1 LOCAL  
390      0567 1     FREE_BNR : REF BBLOCK,  
391      0568 1     LENGTH,  
392      0569 1     STATUS;  
393      0570 1 CNF$TRACE (DBGSC TRACE, $DESCRIPTOR('TRACE'));  
394      0571 1     $DESCRIPTOR ('CNFS CLOSE REQUEST LINK'));  
395      0572 1 IF .IRB [IRBSW_CHAN] NEQ 0          ! If channel was assigned,  
396      0573 1 THEN  
397      0574 1     BEGIN  
398      0575 1     STATUS = $DASSGN (CHAN = .IRB [IRBSW_CHAN]); ! Deassign network channel  
399      0576 1  
400      0577 1 IF NOT .STATUS                      ! If error detected,  
401      0578 1 THEN  
402      0579 1     SIGNAL (CNFS_CHAN, 0, .STATUS); ! then report error  
403      0580 1 END;  
404  
405      0581 1 REMQUE (.IRB, STATUS);           ! Remove from linked list  
406  
407      0582 1  
408      0583 1  
409      0584 1  
410      0585 1  
411      0586 1  
412      0587 1  
413      0588 1  
414      0589 1  
415      0590 1  
416      0591 1  
417      0592 1  
418      0593 1  
419      0594 1  
420      0595 1  
421      0596 1  
422      0597 1  
423      0598 1  
424      0599 1  
425      0600 1  
426      0601 1  
427  
428      0590 1  
429      0591 1  
430      0592 1  
431      0593 1  
432      0594 1  
433      0595 1  
434      0596 1  
435      0597 1  
436      0598 1  
437      0599 1  
438      0600 1  
439      0601 1  
440      0590 1  
441      0591 1  
442      0592 1  
443      0593 1  
444      0594 1  
445      0595 1  
446      0596 1  
447      0597 1  
448      0598 1  
449      0599 1  
450      0600 1  
451      0601 1  
452  
453      0590 1  
454      0591 1  
455      0592 1  
456      0593 1  
457      0594 1  
458      0595 1  
459      0596 1  
460      0597 1  
461      0598 1  
462      0599 1  
463      0600 1  
464      0601 1  
465  
466      0590 1  
467      0591 1  
468      0592 1  
469      0593 1  
470      0594 1  
471      0595 1  
472      0596 1  
473      0597 1  
474      0598 1  
475      0599 1  
476      0600 1  
477      0601 1  
478  
479      0590 1  
480      0591 1  
481      0592 1  
482      0593 1  
483      0594 1  
484      0595 1  
485      0596 1  
486      0597 1  
487      0598 1  
488      0599 1  
489      0600 1  
490      0601 1  
491  
492      0590 1  
493      0591 1  
494      0592 1  
495      0593 1  
496      0594 1  
497      0595 1  
498      0596 1  
499      0597 1  
500      0598 1  
501      0599 1  
502      0600 1  
503      0601 1  
504  
505      0590 1  
506      0591 1  
507      0592 1  
508      0593 1  
509      0594 1  
510      0595 1  
511      0596 1  
512      0597 1  
513      0598 1  
514      0599 1  
515      0600 1  
516      0601 1  
517  
518      0590 1  
519      0591 1  
520      0592 1  
521      0593 1  
522      0594 1  
523      0595 1  
524      0596 1  
525      0597 1  
526      0598 1  
527      0599 1  
528      0600 1  
529      0601 1  
530  
531      0590 1  
532      0591 1  
533      0592 1  
534      0593 1  
535      0594 1  
536      0595 1  
537      0596 1  
538      0597 1  
539      0598 1  
540      0599 1  
541      0600 1  
542      0601 1  
543  
544      0590 1  
545      0591 1  
546      0592 1  
547      0593 1  
548      0594 1  
549      0595 1  
550      0596 1  
551      0597 1  
552      0598 1  
553      0599 1  
554      0600 1  
555      0601 1  
556  
557      0590 1  
558      0591 1  
559      0592 1  
560      0593 1  
561      0594 1  
562      0595 1  
563      0596 1  
564      0597 1  
565      0598 1  
566      0599 1  
567      0600 1  
568      0601 1  
569  
570      0590 1  
571      0591 1  
572      0592 1  
573      0593 1  
574      0594 1  
575      0595 1  
576      0596 1  
577      0597 1  
578      0598 1  
579      0599 1  
580      0600 1  
581      0601 1  
582  
583      0590 1  
584      0591 1  
585      0592 1  
586      0593 1  
587      0594 1  
588      0595 1  
589      0596 1  
590      0597 1  
591      0598 1  
592      0599 1  
593      0600 1  
594      0601 1  
595  
596      0590 1  
597      0591 1  
598      0592 1  
599      0593 1  
600      0594 1  
601      0595 1  
602  
603      0590 1  
604      0591 1  
605      0592 1  
606      0593 1  
607      0594 1  
608      0595 1  
609      0596 1  
610      0597 1  
611      0598 1  
612      0599 1  
613      0600 1  
614      0601 1  
615  
616      0590 1  
617      0591 1  
618      0592 1  
619      0593 1  
620      0594 1  
621      0595 1  
622      0596 1  
623      0597 1  
624      0598 1  
625      0599 1  
626      0600 1  
627      0601 1  
628  
629      0590 1  
630      0591 1  
631      0592 1  
632      0593 1  
633      0594 1  
634      0595 1  
635      0596 1  
636      0597 1  
637      0598 1  
638      0599 1  
639      0600 1  
640      0601 1  
641  
642      0590 1  
643      0591 1  
644      0592 1  
645      0593 1  
646      0594 1  
647      0595 1  
648      0596 1  
649      0597 1  
650      0598 1  
651      0599 1  
652      0600 1  
653      0601 1  
654  
655      0590 1  
656      0591 1  
657      0592 1  
658      0593 1  
659      0594 1  
660      0595 1  
661      0596 1  
662      0597 1  
663      0598 1  
664      0599 1  
665      0600 1  
666      0601 1  
667  
668      0590 1  
669      0591 1  
670      0592 1  
671      0593 1  
672      0594 1  
673      0595 1  
674      0596 1  
675      0597 1  
676      0598 1  
677      0599 1  
678      0600 1  
679      0601 1  
680  
681      0590 1  
682      0591 1  
683      0592 1  
684      0593 1  
685      0594 1  
686      0595 1  
687      0596 1  
688      0597 1  
689      0598 1  
690      0599 1  
691      0600 1  
692      0601 1  
693  
694      0590 1  
695      0591 1  
696      0592 1  
697      0593 1  
698      0594 1  
699      0595 1  
700      0596 1  
701      0597 1  
702      0598 1  
703      0599 1  
704      0600 1  
705      0601 1  
706  
707      0590 1  
708      0591 1  
709      0592 1  
710      0593 1  
711      0594 1  
712      0595 1  
713      0596 1  
714      0597 1  
715      0598 1  
716      0599 1  
717      0600 1  
718      0601 1  
719  
720      0590 1  
721      0591 1  
722      0592 1  
723      0593 1  
724      0594 1  
725      0595 1  
726      0596 1  
727      0597 1  
728      0598 1  
729      0599 1  
730      0600 1  
731      0601 1  
732  
733      0590 1  
734      0591 1  
735      0592 1  
736      0593 1  
737      0594 1  
738      0595 1  
739      0596 1  
740      0597 1  
741      0598 1  
742      0599 1  
743      0600 1  
744      0601 1  
745  
746      0590 1  
747      0591 1  
748      0592 1  
749      0593 1  
750      0594 1  
751      0595 1  
752      0596 1  
753      0597 1  
754      0598 1  
755      0599 1  
756      0600 1  
757      0601 1  
758  
759      0590 1  
760      0591 1  
761      0592 1  
762      0593 1  
763      0594 1  
764      0595 1  
765      0596 1  
766      0597 1  
767      0598 1  
768      0599 1  
769      0600 1  
770      0601 1  
771  
772      0590 1  
773      0591 1  
774      0592 1  
775      0593 1  
776      0594 1  
777      0595 1  
778      0596 1  
779      0597 1  
780      0598 1  
781      0599 1  
782      0600 1  
783      0601 1  
784  
785      0590 1  
786      0591 1  
787      0592 1  
788      0593 1  
789      0594 1  
790      0595 1  
791      0596 1  
792      0597 1  
793      0598 1  
794      0599 1  
795      0600 1  
796      0601 1  
797  
798      0590 1  
799      0591 1  
800      0592 1  
801      0593 1  
802      0594 1  
803      0595 1  
804      0596 1  
805      0597 1  
806      0598 1  
807      0599 1  
808      0600 1  
809      0601 1  
810  
811      0590 1  
812      0591 1  
813      0592 1  
814      0593 1  
815      0594 1  
816      0595 1  
817      0596 1  
818      0597 1  
819      0598 1  
820      0599 1  
821      0600 1  
822      0601 1  
823  
824      0590 1  
825      0591 1  
826      0592 1  
827      0593 1  
828      0594 1  
829      0595 1  
830      0596 1  
831      0597 1  
832      0598 1  
833      0599 1  
834      0600 1  
835      0601 1  
836  
837      0590 1  
838      0591 1  
839      0592 1  
840      0593 1  
841      0594 1  
842      0595 1  
843      0596 1  
844      0597 1  
845      0598 1  
846      0599 1  
847      0600 1  
848      0601 1  
849  
850      0590 1  
851      0591 1  
852      0592 1  
853      0593 1  
854      0594 1  
855      0595 1  
856      0596 1  
857      0597 1  
858      0598 1  
859      0599 1  
860      0600 1  
861      0601 1  
862  
863      0590 1  
864      0591 1  
865      0592 1  
866      0593 1  
867      0594 1  
868      0595 1  
869      0596 1  
870      0597 1  
871      0598 1  
872      0599 1  
873      0600 1  
874      0601 1  
875  
876      0590 1  
877      0591 1  
878      0592 1  
879      0593 1  
880      0594 1  
881      0595 1  
882      0596 1  
883      0597 1  
884      0598 1  
885      0599 1  
886      0600 1  
887      0601 1  
888  
889      0590 1  
890      0591 1  
891      0592 1  
892      0593 1  
893      0594 1  
894      0595 1  
895      0596 1  
896      0597 1  
897      0598 1  
898      0599 1  
899      0600 1  
900      0601 1  
901  
902      0590 1  
903      0591 1  
904      0592 1  
905      0593 1  
906      0594 1  
907      0595 1  
908      0596 1  
909      0597 1  
910      0598 1  
911      0599 1  
912      0600 1  
913      0601 1  
914  
915      0590 1  
916      0591 1  
917      0592 1  
918      0593 1  
919      0594 1  
920      0595 1  
921      0596 1  
922      0597 1  
923      0598 1  
924      0599 1  
925      0600 1  
926      0601 1  
927  
928      0590 1  
929      0591 1  
930      0592 1  
931      0593 1  
932      0594 1  
933      0595 1  
934      0596 1  
935      0597 1  
936      0598 1  
937      0599 1  
938      0600 1  
939      0601 1  
940  
941      0590 1  
942      0591 1  
943      0592 1  
944      0593 1  
945      0594 1  
946      0595 1  
947      0596 1  
948      0597 1  
949     
```

```
: 421      0602 2
: 422      0603 1 END;
:           ! End routine cnf$close_request_link
```

```
.PSECT $PLITS,NOWRT,NOEXE,2
45 43 41 52 54 001AC P.ABF: .ASCII \TRACE\
                                .BLKB 3
                                .LONG 5
                                .ADDRESS P.ABF
45 55 51 45 52 5F 45 53 4F 4C 43 24 46 4E 43 4C 5F 54 53 001BC P.ABH: .ASCII \CNFS CLOSE_REQUEST_LINK\
                                .BLKB 2
                                .LONG 22
                                .ADDRESS P.ABH
                                .EXTRN SYSSDASSGN
.PSECT $CODES,NOWRT,2
54 5E 0000G 001C 00000
                                .ENTRY CNFS CLOSE REQUEST_LINK, Save R2,R3,R4
                                MOVAB CNFSFREE_VM, R4
                                SUBL2 #12, SP
                                PUSHAB P.ABG
                                PUSHAB P.ABE
                                PUSHL #1
                                CALLS #3, CNFSTRACE
                                MOVL IRB, R2
                                TSTW 10(R2)
                                BEQL 1S
                                CVTWL 10(R2), -(SP)
                                CALLS #1, SYSSDASSGN
                                MOVL R0, STATUS
                                BLBS STATUS, 1S
                                PUSHL STATUS
                                CLRL -(SP)
                                PUSHL #CNFS CHAN
                                CALLS #3, LIB$SIGNAL
                                REMQUE (R2), STATUS
                                MOVL IRB, R0
                                MOVL 20(R0), FREE_BNR
                                ADDL3 #20, IRB, R2
                                CMPL FREE_BNR, R2
                                BEQL 3S
                                REMQUE #FREE_BNR, STATUS
                                ADDL3 #12, FREE_BNR, -(SP)
                                ADDL3 #8, FREE_BNR, -(SP)
                                CALLS #2, CNFSFREE_VM
                                BLBC STATUS, 4S
                                PUSHAB FREE_BNR
                                MOVL #16-4(SP)
                                PUSHAB 4(SP)
                                CALLS #2, CNFSFREE_VM
                                BLBC STATUS, 4S
                                ADDL3 #20, IRB, R2
                                MOVL (R2), FREE_BNR
: 0546
: 0575
: 0574
: 0577
: 0580
: 0582
: 0584
: 0587
: 0592
: 0593
: 0595
: 0596
: 0597
: 0598
```

CNFINTRPT  
V04-000

DECnet Ethernet Configurator Module  
CNFSCLOSE\_REQUEST\_LINK

10  
16-Sep-1984 02:03:38  
14-Sep-1984 12:49:50  
VAX-11 Bliss-32 V4.0-742  
[NICNF.SRC]CNFINTRPT.B32;1

Page 16  
(6)

08	50	04	CB	11	00088	BRB	2\$		0593
	AE	08	AC	00	0008A	MOVL	IRB	RO	0600
		04	AO	32	0008E	CVTWL	8(R0),	LENGTH	0601
		0C	AC	9F	00093	PUSHAB	IRB		
			AE	9F	00096	PUSHAB	LENGTH		
64		02	FB	00099	CALLS	#2,	CNF\$FREE_VM		
		04	0009C	48:	RET				0603

; Routine Size: 157 bytes, Routine Base: \$CODE\$ + 01B9

```

: 424      0604 1 XSBTTL 'shutdown'
: 425      0605 1 ROUTINE SHUTDOWN: NOVALUE =
: 426      0606 1
: 427      0607 1 ---  

: 428      0608 1 This routine is called when the network is shutting down to
: 429      0609 1 gracefully close all incoming links so that NICONFIG goes away quietly.
: 430      0610 1
: 431      0611 1 Inputs:
: 432      0612 1     None
: 433      0613 1
: 434      0614 1 Outputs:
: 435      0615 1     None
: 436      0616 1
: 437      0617 1 Effect:
: 438      0618 1     Disabling all surveillance will cause NICONFIG to terminate
: 439      0619 1
: 440      0620 1 ---  

: 441      0621 1 BEGIN
: 442      0622 1 LOCAL
: 443      0623 1     PTR:     REF BBLOCK,           ! Pointer to irb block
: 444      0624 1     NEXT_PTR;
: 445      0625 1 PTR = .CNF$GQ_IRBLST;          ! Start at first link context block
: 446      0626 1 WHILE .PTR NEQ CNF$GQ_IRBLST   ! Until end of linked list.
: 447      0627 1 DO
: 448      0628 1     BEGIN
: 449      0629 1     NEXT_PTR = .PTR [IRBSL_LINK];
: 450      0630 1     CNF$CLOSE REQUEST_LINKT,PTR); ! Abort the incoming link
: 451      0631 1     PTR = .NEXT_PTR;           ! and link to next in chain
: 452      0632 1 END;  

: 453      0633 1 PTR = .CNF$GQ_CIRSLURLST;       ! Start at first circuit block
: 454      0634 1 WHILE .PTR NEQ CNF$GQ_CIRSLURLST ! Until end of linked list.
: 455      0635 1 DO
: 456      0636 1     BEGIN
: 457      0637 1     CNF$DISABLE SURVEIL (.PTR); ! and delete the circuit
: 458      0638 1     PTR = .PTR [CIRSL_LINK];    ! Link to next one
: 459      0639 1 END;  

: 460      0640 1 CNF$B_SURVEILLANCE_SET = FALSE; ! So it will die quietly
: 461      0641 1
: 462      0642 1 END;                           ! End routine shutdown
: 463
: 464
: 465
: 466
: 467
: 468
: 469
: 470

```

## 000C 00000 SHUTDOWN:

S2	0000G	CF	D0	00002	.WORD	Save R2,R3	: 0605
50	0000G	CF	9E	00007 1\$:	MOVL	CNF\$GQ_IRBLST, PTR	: 0631
50		52	D1	0000C	MOVAB	CNF\$GQ_IRBLST, R0	: 0632
					CMPL	PTR, R0	

CNFINTRPT  
V04-000

DECnet Ethernet Configurator Module  
shutdown

K 10  
16-Sep-1984 02:03:38  
14-Sep-1984 12:49:50

VAX-11 Blfss-32 V4.0-742  
[NICNF.SRC]CNFINTRPT.B32;1

Page 18  
(7)

CN  
VO

			0F	13	0000F	BEQL	2\$				
			62	DD	00011	MOVL	(PTR), NEXT_PTR				0635
			52	DD	00014	PUSHL	PTR				0636
	FF48	CF	01	FB	00016	CALLS	#1, CNFSCLOSE_REQUEST_LINK				
		52	53	DD	0001B	MOVL	NEXT_PTR, PTR				0637
			E7	11	0001E	BRB	1\$				0632
		52	CF	DD	00020	2\$:	MOVL	CNFSGQ_CIRSURLST, PTR			0640
		50	CF	9F	00025	3\$:	MOVAB	CNFSGQ_CIRSURLST, R0			0641
		50	52	D1	0002A	CMPL	PTR, R0				
			0C	13	0002D	BEQL	4\$				
			52	DD	0002F	PUSHL	PTR				0644
	0000G	CF	01	FB	00031	CALLS	#1, CNFSDISABLE_SURVEIL				
		52	62	DD	00036	MOVL	(PTR), PTR				0645
			EA	11	00039	BRB	3\$				0641
		0000G	CF	D4	0003B	4\$:	CLRL	CNFSB_SURVEILLANCE_SET			0648
			04	0003F		RET					0650

; Routine Size: 64 bytes, Routine Base: \$CODES + 0256

```

672      0651 1 %SBTTL 'CNFS$SOLICIT REQUEST'
673      0652 1 GLOBAL ROUTINE CNFS$SOLICIT_REQUEST (irb): NOVALUE =
674
675      0653 1 !---
676      0654 1
677      0655 1 This routine is called to obtain requests from the incoming
678      0656 1 logical link. Each incoming request is immediately queued
679      0657 1 to the disposal queue for the appropriate action.
680      0658 1 It is first called directly by OPEN REQUEST_LINK which is executing
681      0659 1 off the work queue, and thereafter calls are placed on the work queue
682      0660 1 by the AST routine, REQUEST_RECEIVED.
683      0661 1
684      0662 1
685      0663 1 Inputs:
686      0664 1     irb = Address of incoming request block
687      0665 1
688      0666 1 Outputs:
689      0667 1     None
690      0668 1
691      0669 1 !---
692      0670 1
693      0671 2 BEGIN
694      0672 2
695      0673 2 MAP
696      0674 2     IRB:      REF BBLOCK;           ! Address of incoming request block
697      0675 2
698      0676 2 LOCAL
699      0677 2     STATUS;
700      0678 2
701      0679 2
702      0680 2 CNF$TRACE (DBGSC TRACE, $DESCRIPTOR('TRACE'),
703      0681 2             $DESCRIPTOR ('cnf$solicit_request'));
704      0682 2
705      P 0683 2 STATUS = $QIO(FUNC = IOS_READVBLK,          ! Get request from incoming link
706      P 0684 2             CHAN = .IRB [IRBSW_CHAN],
707      P 0685 2             EFN = CNFSC ASYNCH_EFN,
708      P 0686 2             IOSB = IRB [IRBSW_IOSB], ! Address of I/O status block
709      P 0687 2             ASTADR = REQUEST_RECEIVED, ! Address of completion routine
710      P 0688 2             ASTPRM = IRB,           ! Giving irb as routine parameter
711      P 0689 2             P1 = IRB [IRBS REQUEST], ! Address of request buffer
712      P 0690 2             P2 = IRBS$C_MAXREQUESTLEN); ! Length of request buffer
713      0691 2
714      0692 2 IF NOT .STATUS                         ! If unsuccessful
715      0693 2 THEN
716      0694 2     BEGIN
717      0695 2         IF (.STATUS NEQ SSS_LINKABORT) AND ! Don't signal a fatal error just because
718      0696 2             (.STATUS NEQ SSS_LINKEXIT) ! the partner went away.
719      0697 2         THEN
720      0698 2             SIGNAL (CNFS_LINK, 0, .STATUS); ! then report error
721      0699 2
722      0700 2 WQSADD WORK ITEM( CNF$CLOSE REQUEST_LINK, IRB);
723      0701 2 CNF$TRACE (DBGSC TRACE, $DESCRIPTOR('TRACE'),
724      0702 2             $DESCRIPTOR ('cnf$solicit_request' %%CNF$CLOSE_REQUEST_LINK'));
725      0703 2
726      0704 2 END;
727      0705 1 END;                                ! End routine cnf$solicit_request

```

							.PSECT SPLIT\$, NOWRT, NOEXE, 2											
							.ASCII \TRACE\											
							.BLKB 3											
							.LONG 5											
							.ADDRESS P.ABJ											
71	65	72	5F	74	69	63	69	6C	6F	73	24	66	6E	63	001DC	P.ABJ:	.ASCII \cnf\$solicit_request\	
															001E1			
															001E4	P.ABI:	.LONG 5	
															001E8			
															001EC	P.ABL:	.BLKB 3	
															001FB			
															001FF			
															00200	P.ABK:	.BLKB 1	
															00204			
															00208	P.ABN:	.LONG 19	
															0020D			
															00210	P.ABM:	.ADDRESS P.ABL	
															00214		.ASCII \TRACE\	
															00218	P.ABP:	.BLKB 3	
															00227		.LONG 5	
															00236		.ADDRESS P.ABM	
															00240		.ASCII \cnf\$solicit_request %CNFS CLOSE REQUEST\	
															00245		.BLKB 3	
															00248	P.ABO:	.LONG 45	
															0024C		.ADDRESS P.ABP	

							.PSECT SCODE\$, NOWRT, 2	
							.ENTRY CNFSOLICIT_REQUEST, Save R2	
							PUSHAB P.ABK	
							PUSHAB P.ABI	
							PUSHL #1	
							CALLS #3, CNF\$TRACE	
							CLRQ -(SP)	
							CLRQ -(SP)	
							MOVZBL #250, -(SP)	
							MOVL IRB, R2	
							PUSHAB 101(R2)	
							PUSHL R2	
							PUSHAB REQUEST_RECEIVED	
							PUSHAB 12(R2)	
							PUSHL #49	
							CVTWL 10(R2), -(SP)	
							PUSHL #CNFS ASYNCH_EFN	
							CALLS #12, SYS\$Q10	
							BLBS STATUS, 28	
							CMPL STATUS, #8420	
							BEQL 1\$	
							CMPL STATUS, #8436	
							BEQL 1\$	
							PUSHL STATUS	
							CLRL -(SP)	
							PUSHL #CNFS LINK	
							CALLS #3, LIB\$SIGNAL	
							PUSHL R2	
							PUSHAB CNFS CLOSE REQUEST LINK	
							CALLS #2, WK\$ADD_WORK_ITEM	

CNFINTRPT  
V04-000

DECnet Ethernet Configurator Module  
CNFSOLICIT\_REQUEST

N 10

16-Sep-1984 02:03:38  
14-Sep-1984 12:49:50

VAX-11 Bliss-32 V4.0-742  
[NICNF.SRC]CNFINTRPT.B32;1

Page 21  
(8)

0000: CF 9F 0006D  
0000: CF 9F 00071  
0000G CF 01 DD 00075  
0000G CF 03 FB 00077  
0000G CF 04 0007C 28:  
PUSHAB P.ABO  
PUSHAB P.ABM  
PUSHL #1  
CALLS #3, CNF\$TRACE  
RET

: 0702  
: 0701  
: 0705

; Routine Size: 125 bytes. Routine Base: \$CODE\$ + 0296

```
528      0706 1 ISBTTL 'request_received'  
529      0707 1 ROUTINE REQUEST_RECEIVED (irb): NOVALUE =  
530      0708 1 ---  
531      0709 1 This AST routine is called when a new request has come  
532      0710 1 in over the logical link. The request is queued to the  
533      0711 1 work queue.  
534      0712 1  
535      0713 1 Inputs:  
536      0714 1  
537      0715 1      irb = Address of incoming request block  
538      0716 1  
539      0717 1 Outputs:  
540      0718 1      None  
541      0719 1  
542      0720 1  
543      0721 1  
544      0722 1  
545      0723 1  
546      0724 2 BEGIN  
547      0725 2  
548      0726 2 MAP  
549      0727 2      IRB:      REF BBLOCK;          ! Address of incoming request block  
550      0728 2  
551      0729 2 CNF$TRACE (DBGSC_TRACE, $DESCRIPTOR('TRACE'),  
552      0730 2           $DESCRIPTOR ('request_received'));  
553      0731 2  
554      0732 2 IF NOT .IRB [IRBSW_IOSB]          ! If error from QIO,  
555      0733 2 THEN  
556      0734 3 BEGIN  
557      0735 3      IF (.IRB [IRBSW_IOSB] NEQ SSS_LINKABORT) AND  
558      0736 4           (.IRB [IRBSW_IOSB] NEQ SSS_LINKEXIT)  
559      0737 4 THEN  
560      0738 4           SIGNAL (CNFS_LINK, 0, .IRB [IRBSW_IOSB]);    ! then report the error  
561      0739 4           WKQSADD_WORK_ITEM( CNFS CLOSE_REQUEST_LINK, .IRB);  ! close the link until re-established  
562      0740 4 RETURN;  
563      0741 2 END;  
564      0742 2  
565      0743 2  
566      0744 2      Log the contents of the incoming message  
567      0745 2  
568      0746 2 BEGIN  
569      0747 2 LOCAL DATA_DSC : BBLOCK [DSC$C_S_BLN];  
570      0748 2  
571      0749 2 DATA_DSC = 0;  
572      0750 2 DATA_DSC [DSC$W_LENGTH] = .IRB [IRBSW_IOSB1];  
573      0751 2 DATA_DSC [DSC$A_POINTER] = IRB [IRBS$REQUEST];  
574      0752 2 CNFS LOG_DATA (DBGSC_NICE, $DESCRIPTORT'NICE received'),0, DATA_DSC);  
575      0753 2 END;  
576      0754 2  
577      0755 2 WKQSADD WORK_ITEM (CNFS PROCESS REQUEST, ! Queue request  
578      0756 2           .IRB);  
579      0757 2  
580      0758 1 END;          ! End routine request_received
```

.PSECT SPLIT\$,NOWRT,NOEXE,2

	45 43 41 52 54	00250 P.ABR: .ASCII \TRACE\	:
	00000005	00255 .BLKB 3	:
	00000000	00258 P.ABQ: .LONG 5	:
65 76 69 65 63 65 72 5F 74 73 65 75 71 65 72 64	0025C .ADDRESS P.ABR	:	
	00000010	00260 P.ABT: .ASCII \request_received\	:
	00000000	0026F .P.ABS: .LONG 16	:
64 65 76 69 65 63 65 72 20 45 43 49 4E	00274 .ADDRESS P.ABT	:	
	0000000D	00278 P.ABV: .ASCII \NICE received\	:
	00000000	00285 .BLKB 3	:
	00288 P.ABU: .LONG 13	:	
	0028C .ADDRESS P.ABV	:	

## .PSECT \$CODES,NOWRT,2

0004 00000 REQUEST_RECEIVED:							
							.WORD Save R2
	5E	0000:	08 C2 00002				SUBL2 #8, SP
		0000:	CF 9F 00005				PUSHAB P.ABS
			01 DD 0000D				PUSHAB P.ABQ
			03 FB 0000F				PUSHL #1
0000G	CF	04	AC D0 00014				CALLS #3, CNF\$TRACE
	52	04	OC A2 E8 00018				MOVL IRB, R2
	2B	0C	A2 B1 0001C				BLBS 12(R2), 28
20E4	8F	0C	1B 13 00022				CMPW 12(R2), #8420
			13 13 00024				BEQL 1\$
20F4	8F	0C	A2 B1 00024				CMPW 12(R2), #8436
			13 13 0002A				BEQL 1\$
	7E	3C	A2 32 0002C				CVTWL 12(R2), -(SP)
		7E	D4 00030				CLRL -(SP)
		0000000G	8F DD 00032				PUSHL #CNFS LINK
0000000G	00	0000000G	03 FB 00038				CALLS #3, LIB\$SIGNAL
			52 DD 0003F	1\$:			PUSHL R2
		FE61	CF 9F 00041				CALLS CNF\$CLOSE_REQUEST_LINK
			20 11 00045				PUSHAB 3\$
			6E D4 00047	2\$:			BRB DATA_DSC
04	6E	0E	A2 B0 00049				CLRL 14(R2), DATA_DSC
	AE	65	A2 9E 0004D				MOVW 101(R2), DATA_DSC+4
			5E DD 00052				MOVAB SP
			7E D4 00054				PUSHL -(SP)
		0000:	CF 9F 00056				CLRL P.ABU
0000G	CF	04	FB 0005C				CLRL -(SP)
		0000G	52 DD 00061				CALLS #4, CNF\$LOG_DATA
0000G	CF	02	FB 00063	3\$:			PUSHL R2
		04 00067	04 0006C				PUSHAB CNF\$PROCESS_REQUEST
							CALLS #2, WKQSADD_WORK_ITEM
							RET

; Routine Size: 109 bytes, Routine Base: \$CODE\$ + 0313

CNFINTRPT  
VO4-000

DECnet Ethernet Configurator Module  
request\_received

D 11  
16-Sep-1984 02:03:38 VAX-11 Bliss-32 V4.0-742  
14-Sep-1984 12:49:50 [NICNF.SRC]CNFINTRPT.B32;1

Page 24  
(10)

: 582 0759 1 END  
: 583 0760 0 ELUDOM

: ! End of module CNFINTRPT

.EXTRN LIB\$SIGNAL

PSECT SUMMARY

Name

Bytes

Attributes

\$OWNS	8 NOVEC, WRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
SPLITS	656 NOVEC,NOWRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
SCODES	896 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File

Total	Symbols
Loaded	Percent

Pages
Mapped

Processing
Time

-\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	16	0	581	00:01.0
-\$255\$DUA28:[SHRLIB]NET.L32;1	1279	0	0	63	00:00.9
-\$255\$DUA28:[SHRLIB]NMALIBRY.L32;1	887	0	0	47	00:00.8

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:\$CNFINTRPT/OBJ=OBJ\$:\$CNFINTRPT MSRC\$:\$CNFINTRPT/UPDATE=(ENH\$:\$CNFINTRPT)

: Size: 896 code + 664 data bytes  
: Run Time: 00:19.2  
: Elapsed Time: 00:41.8  
: Lines/CPU Min: 2370  
: Lexemes/CPU-Min: 20292  
: Memory Used: 116 pages  
: Compilation Complete

0279 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

NICNF

CNFDEF  
SOL

CNFDEF  
LIS

NETTRN  
LIS

NICONFIG  
MAP

NETTREE  
LIS

CNFMAIN  
LIS

CNFREQES  
LIS

SERVER  
LIS

CNFINTRPT  
LIS

CNFWDDEF  
SOL

CNFPREFIX  
REQ

CNFMSG  
LIS